

- Cada pregunta de la múltiple opción contestada correctamente tiene un valor de 3,34 puntos. Cada pregunta incorrecta de la múltiple opción resta 1,11 puntos.
- Para aprobar el examen se deben obtener al menos 60 puntos.

Parte 1 Múltiple Opción

1) Dado el siguiente código

```
private double procesarValor(String strValor) {
    double valor;
    try {
        valor = Double.parseDouble(strValor);
        if (valor > 999.99) {
            valor = 0;
        }
    } catch (NumberFormatException ex) {
        valor = 0;
    }
    return valor;
}
```

y dados los siguientes conjuntos de datos de prueba para la variable strValor:

- I) {"999.99","0"}
- II) {"0","abc","-38"}
- III) {"-39","0","1000"}

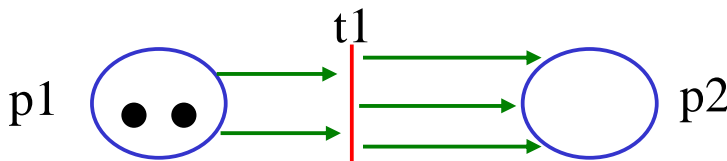
Se cumple que:

- a. La unión de los conjuntos I y II cumple con el criterio de cubrimiento de sentencias
 - b. III cumple con el criterio de cubrimiento de sentencias
 - c. **La unión de los conjuntos II y III cumple con el criterio de cubrimiento de sentencias.**
 - d. Ninguna es verdadera
-
- 2) Un programa falla cuando:
- a. No hace lo requerido o hace algo que no debería hacer.
 - b. a) y posibles razones de la falla son defectos en el diseño
 - c. **b) y posibles razones de la falla son defectos en el código**
 - d. Ninguna de las anteriores
-
- 3) En el proceso de Verificación y Validación las técnicas dinámicas de verificación
- a. **Se basan en experimentar con el producto de software para identificar diferencias entre el comportamiento real y el esperado.**
 - b. Se basan en analizar el producto para deducir su correcta operación.
 - c. b) y sirven para verificar no solo el código sino también los requerimientos y el diseño.
 - d. c) y son más efectivas que las técnicas estáticas.
-
- 4) Sobre las pruebas de integración
- a. La estrategia big-bang simplifica la detección del origen de las fallas detectadas respecto a estrategias incrementales de integración
 - b. **La estrategia big-bang realiza una integración en la cual los módulos no se van agregando de forma incremental sino que se integran todos a la vez.**
 - c. b) y tanto la estrategia bottom-up como la top-down necesitan de drivers y de stubs para utilizarse
 - d. La estrategia "first come integration" integra los módulos a medida que estos se terminan de desarrollar, por ende, esta estrategia es difícil de planificar
-
- 5)Cuál de estos criterios es el menos apropiado si se usa como único criterio para terminar las pruebas
- a. **Terminar las pruebas cuando el tiempo establecido para las mismas ha terminado**
 - b. Terminar las pruebas cuando se ha detectado cierta cantidad de defectos de los defectos totales estimados y no se detectan fallas graves
 - c. Terminar las pruebas basándose en criterios pre-establecidos de diseño de casos de prueba (y que estos casos de prueba se ejecuten correctamente)
 - d. Todos estos criterios son apropiados si se usan de forma independiente

-
- 6) La arquitectura de software para un sistema
- es la estructura o estructuras del sistema, la cual consiste en elementos, sus propiedades externamente visibles, y las relaciones entre ellos.
 - a) y define el proceso que se va a utilizar durante todo el desarrollo del software
 - es un proceso en cascada
 - Ninguna de las anteriores
-
- 7) Documentar la arquitectura de software es una ventaja porque:
- Contribuye a la comunicación entre stakeholders
 - Facilita la toma temprana de decisiones
 - Promueve el reuso a gran escala
 - Todas las anteriores
-
- 8) En una arquitectura orientada a servicios, un servicio web
- es independiente de las aplicaciones que lo usan
 - a) y el lenguaje de desarrollo a utilizar para los mismos es Java
 - b) utilizando la API Java Service Oriented Connection
 - Ninguna de las anteriores
-
- 9)
- Los estilos (de arquitecturas de software) no son usados en la práctica ya que es casi imposible que un sistema se comporte exactamente como un estilo
 - Los estilos (de arquitecturas de software) son un marco teórico sobre el cual luego se construyen los llamados "estilos prácticos". Siendo estos últimos los que se utilizan en la construcción de sistemas de software.
 - El estilo (de arquitecturas de software) de nombre capas jerárquicas no admite que se haga una invocación a una capa que no es la inmediatamente "inferior" (considerar que las capas superiores llaman a capas inferiores)
 - Ninguna es correcta
-
- 10) Cuando se evalúan los resultados obtenidos luego de la aplicación de un proceso, se está evaluando:
- La madurez de la organización.
 - El desempeño del proceso
 - La capacidad de un proceso
 - a y c
-
- 11)
- La "Programación de a Pares" es un método de programación que involucra a 2 personas trabajando juntas durante la codificación
 - a) y una persona codifica mientras que la otra realiza una revisión de código en tiempo real
 - b) y al utilizar este método la productividad aumenta en más o menos un 50% debido al bajo porcentaje de retrabajo (que es alcanzado mediante la alta calidad del código desarrollado)
 - Ninguna de las anteriores
-
- 12) La estructura de las áreas de proceso del modelo CMMI:
- Varía para las distintas categorías de áreas de proceso
 - Propone metas genéricas y metas específicas para cada área de proceso
 - b) y estas metas son requeridas a diferencia de las prácticas que son esperadas
 - Ninguna de las anteriores
-
- 13) En el marco de la gestión de proyectos se define el concepto de programa. Un programa se utiliza para:
- por ejemplo: subcontratar a un proveedor.
 - Manejar conjuntamente proyectos relacionados.
 - Manejar proyectos no necesariamente relacionados.
 - Ninguna de las anteriores
-
- 14) La disciplina de gestión de proyectos conlleva
- Equilibrar demandas de tiempo, costo, alcance y calidad.
 - Tener en cuenta las restricciones existentes de tecnología, personas y procesos.
 - Satisfacer las necesidades y expectativas de los clientes y stakeholders
 - Todas las anteriores

- 15) En la planificación de un proyecto un WBS se utiliza para
- Organizar y definir el alcance total del proyecto.
 - Identificar las actividades del camino crítico del proyecto.
 - a) y permite identificar los entregables del proyecto**
 - b) e identificar los recursos necesarios para llevar adelante cada actividad.
-
- 16) Para realizar estimaciones en un proyecto, se pueden usar diversas técnicas. Una de ellas consiste en estimar a partir de los datos históricos de la organización. Con respecto a esta técnica:
- es adecuado utilizarla para realizar estimaciones en proyectos pequeños debido a que en éstos, las características de la organización influyen más que las capacidades personales
 - los datos históricos están influidos por la subjetividad del estimador
 - no es tan buena como el juicio de expertos
 - lleva implícita la asunción simplificadora de que las cosas van a ir más o menos igual que en proyectos anteriores.**
-
- 17) Los requisitos describen:
- Las funcionalidades y servicios del sistema a construir
 - La interacción del sistema con su entorno
 - Las restricciones que limitan la elección de la solución
 - Todos los anteriores**
-
- 18) Los casos de uso:
- son una técnica para entender y describir requisitos
 - a), con la limitación que no es útil para representar requisitos no funcionales.
 - b) y especifican qué es lo que el sistema debe hacer, sin especificar cómo debe hacerlo**
 - Es la técnica mas adecuada para describir procesos *batch*
-
- 19) El modelo de procesos genérico de obtención, análisis y documentación de requisitos puede verse como:
- Un modelo en cascada, donde las actividades se dan en sucesión.
 - Un modelo en espiral, donde las actividades se desarrollan de forma iterativa incremental.**
 - Un modelo distribuido, donde cada actividad es realizada por un equipo diferente.
 - Actividades que pueden realizarse en cualquier orden

- 20) Una red de Petri con la siguiente configuración



- está habilitada**
 - no está habilitada porque se necesitaría que hubieran 3 marcas en p1 para poder disparar t1
 - Consume una marca de p1 cada vez que se dispara
 - Coloca dos marcas en p2 al dispararse t1
-
- 21) El objetivo de la actividad validación de requisitos, es asegurarse que:
- el documento de especificación de requerimientos cumple con los estándares de la organización
 - el sistema que se está construyendo, efectivamente construye lo que especifican los requisitos
 - los requisitos no funcionales relevados son razonables
 - los requisitos relevados se corresponden con las necesidades del cliente**
-
- 22) Indique cuál de estas afirmaciones es verdadera:
- La creciente importancia y dependencia del software en la sociedad actual está trayendo aparejados un creciente reclamo por niveles de calidad adecuados y una proliferación de normas y regulaciones referidas a la producción y uso de software.**
 - A medida que se desarrolla y como resultado de la evolución tecnológica, la ingeniería de software se independiza de cuestiones legales, reglamentarias y éticas.
 - La ingeniería de software pone el énfasis en las características intrínsecas al software, independientemente del hardware en el que funcione y de las personas con las que interactúe.
 - Si consideramos que un producto de software es “correcto” si cumple con su especificación, un producto de software “correcto” será adecuado al uso previsto además de confiable.

23) Entre los atributos de calidad más relevantes para un producto de software que permita realizar y gestionar inscripciones en concursos para trabajar en el estado están:

- adecuación al uso, facilidad de mantenimiento, modularidad
- seguridad de la información (security), facilidad de mantenimiento, facilidad de verificación, facilidad de adaptación
- tiempo de respuesta, adecuación al uso, consumo de recursos, seguridad de la integridad física (safety)
- tiempo de respuesta, adecuación al uso, consumo de recursos, seguridad de la información (security)**

24) Respecto a modelos de proceso de software:

- El esquema ETVX solo es adecuado para especificar modelos de proceso descriptivos.
- Una de las ventajas de un proceso en fases con evaluaciones parciales por usuarios es que permite acortar el plazo para poder contar con resultados de utilidad, lo que a su vez permite atender la cada vez más buscada reducción del plazo para salir al mercado (time-to market).
- La planificación de un proyecto de desarrollo de software es independiente del modelo de proceso que se esté aplicando.
- El contar con un modelo de proceso permite razonar sobre la forma mediante la cual construimos software, en particular para encarar la mejora del proceso, por ejemplo para generar productos de mejor calidad, a menor costo o en menor plazo.**

25) Dados los siguientes proyectos y modelos de proceso.

Proyectos:

A - Alto riesgo, gran tamaño (10.000 Puntos de Función), alcance ambicioso y con aspectos no del todo definidos, al menos un subconjunto de funcionalidad definido que resulta útil para los usuarios, involucra a múltiples organizaciones, cada una con dispersión geográfica y a más de 3000 usuarios, larga duración (dos años o más).

B – Riesgo medio, alcance definido, al menos dos subconjuntos de funcionalidad definidos que resultan útiles para los usuarios, duración media (nueve meses).

C – Riesgo bajo, simple, alcance definido, corta duración (cuatro semanas).

Modelos de Proceso:

- Cascada
- En Fases con Evaluaciones Parciales por usuarios
- En Fases con Liberaciones Parciales en Producción y cada Fase Cascada
- En Fases con Liberaciones Parciales en Producción y cada Fase en Fases con Evaluaciones Parciales por usuarios

Elija la asignación que le parezca más adecuada.

- A1, B1, C1
- A2, B2, C4
- A1, B2, C4
- A4, B3, C1**

26) Al liberar un sistema:

- En general resulta necesario llevar a cabo un entrenamiento del mismo
- a) y en general dicho entrenamiento debe estar dirigido tanto a usuarios finales como a administradores del sistema
- b) y como parte del entrenamiento se pueden plantear demostraciones y talleres, los que a su vez pueden constituir instancias de validación del software**
- Conviene que el entrenamiento se imparta en etapas tempranas del proyecto a todos los usuarios.

27) Respecto a la liberación del software:

- Una explicación de que existan organizaciones que continúan soportando sus funciones críticas mediante un sistema legado (construido hace muchos años y en general con tecnologías hoy en día obsoletas), con costos elevados y crecientes de mantenimiento, es el poco esfuerzo que puede requerir la implantación de un sistema nuevo que lo sustituya.
- Entre las principales ventajas de la estrategia big-bang en el caso de un sistema que sustituye a otro sistema anterior, están que evita los problemas que origina la coexistencia entre ambos sistemas y que se obtienen las ventajas del sistema nuevo desde el comienzo de la liberación.**
- En general conviene comenzar la planificación de la sustitución en producción de un sistema anterior a uno nuevo cuando ya están avanzadas las pruebas a nivel de sistema (del nuevo sistema que se está desarrollando) y se avizora su puesta en explotación
- c) y al definir el alcance de un proyecto debe quedar claro si incluye o no tareas de la liberación del software tales como capacitación, entrenamiento y documentación para usuarios, instalación, soporte y solución de problemas, conversión de datos.

28) El mantenimiento de software se puede clasificar en “correctivo”, “preventivo”, “perfectivo” y “adaptativo”. El esfuerzo que dedica una organización a cada uno de estos distintos tipos de mantenimiento constituye un elemento de información relevante para evaluar la calidad del software bajo mantenimiento y también de la del propio proceso de mantenimiento.

- a. En general es deseable que una organización de mantenimiento dedique la mayor parte de su esfuerzo de mantenimiento al de tipo “correctivo” ya que esta es una vía para lograr obtener un software de mejor calidad.
- b. a) y en general es muy baja la probabilidad de que se introduzca un defecto nuevo como resultado de una intervención de mantenimiento de tipo “correctivo”.
- c. Un caso de mantenimiento de tipo “preventivo” se da cuando en una organización se decide reescribir un componente que presenta una historia de múltiples defectos reportados y ya corregidos a lo largo de un período.
- d. c) y para saber si es conveniente realizar un mantenimiento de ese tipo se requiere contar con un registro de los defectos por componente.

29) Sobre la gestión de la configuración:

- a. El responsable de SCM es quién resuelve si aprobar o no las solicitudes de cambio
- b. En proyectos grandes es importante contar con herramientas que apoyen la gestión de la configuración.
- c. El Comité de Control de Cambios es quién evalúa y aprueba los cambios en los distintos elementos de configuración.
- d. Las opciones (b) y (c) son correctas.

30) Son actividades de SCM:

- a. Definir elementos de configuración y controlar su versionado
- b. a), controlar los cambios en todos los documentos del proyecto
- c. Asegurar la trazabilidad de requerimientos
- d. Ninguna de las anteriores